# MULTIPLE OIL PAD DETECTION USING DEEP LEARNING

Anish Giri[1], Sajith Variyar V V[1], Sowmya V[1], Ramesh Sivanpillai[2], Soman, K.P[1]

[1] Center for Computational Engineering and Networking (CEN), Amrita School of Engineering, Coimbatore,
Amrita Vishwa Vidyapeetham, India - (ORCID:0000-0002-4674-4828)–anishgiri2910@gmail.com,
(ORCID: 0000-0003-3944-8155)–vv_sajithvariyar@cb.amrita.edu, v_sowmy@cb.amrita.edu
[2] Wyoming GIS Center, University of Wyoming, Laramie, WY 82072, USA (ORCID: 0000-0003-3547-9464)
–sivan@uwyo.edu

**KEY WORDS:** Deep Learning, Oil pads, Faster R-CNN, Aerial images, Data Augmentation.

## ABSTRACT:

Deep learning (DL) algorithms are widely used in object detection such as roads, vehicles, buildings, etc., in aerial images. However, the object detection task is still considered challenging for detecting complex structures, oil pads are one such example: due to its shape, orientation, and background reflection. A recent study used Faster Region-based Convolutional Neural Network (FR-CNN) to detect a single oil pad from the center of the image of size 256 x 256. However, for real-time applications, it is necessary to detect multiple oil pads from aerial images irrespective of their orientation. In this study, FR-CNN was trained to detect multiple oil pads. We cropped images from high spatial resolution images to train the model containing multiple oil pads. The network was trained for 100 epochs using 164 training images and tested with 50 images under 3 different categories. with images containing: single oil pad, multiple oil pad and no oil pad. The model performance was evaluated using standard metrics: precision, recall, F1-score. The final model trained for multiple oil pad detection achieved a weighted average for 50 images precision of 0.67, recall of 0.80, and f1 score of 0.73. The 0.80 recall score indicates that 80% of the oil pads were able to identify from the given test set. The presence of instances in test images like cleared areas, rock structures, and sand patterns having high visual similarity with the target resulted in a low precision score.

## 1. INTRODUCTION

In the recent past, deep learning (DL) algorithms are the most popular technique used in computer vision for object detection tasks. However, object detection tasks such as complex man-made structures are still considered to be one of the most challenging tasks for DL algorithms. Many man-made objects appear small in aerial images and often blend with surrounding features, which makes their discrimination challenging. Objects when viewed from a top-down perspective can be orientated facing any angle (Mo and Yan, 2020). Thousands of images are necessary to train most DL networks which could be a challenge for some applications. With the aforementioned challenges and uses in real-life applications, object detection for aerial images is a popular field of research.

There are several DL-based object detection algorithms (Osco et al., 2021), (Jiao et al., 2019) like YOLO9000 (Redmon and Farhadi, 2017) , SSD (Liu et al., 2015), and Faster R-CNN (Ren et al., 2015) that can be used for identifying objects in aerial images. Object detection applications in aerial images include vehicle detection (Ajay et al., 2017) (Mo and Yan, 2020) (Mohan et al., 2018), weed identification (Bah et al., 2018), estimation of the extent of floods from crowd sourced images (Geetha et al., 2017), etc. (Bah et al., 2018) used an unsupervised learning approach with CNN to detect weeds from drone images. The challenges mentioned were the target annotation and the similarity between the weed as target and the crops as background making it difficult for deep learning algorithms to distinguish them. (Mo and Yan, 2020) addressed the issues of vehicles being small in size and having class imbalance issues caused by different numbers of objects in the vehicle classes. The issues were addressed by creating a new dataset, adding more vehicles and stitching them to the images to artificially augment data. With some modifications to the pooling operation and with a joint training loss function modified done for the Faster R-CNN, the results were improved by 8% compared to the original Faster R-CNN network.(Ho et al., 2019) used Faster R-CNN (Girshick et al., 2016) to detect watermelons for estimating the yields. In that study the canvas(background) and the watermelon were very similar, but Faster R-CNN was still able to distinguish the target from the background. So Faster R-CNN is effective in finding targets when background reflection and the reflection from the target is similar.

(Sunil et al., 2021) investigated the potential of Faster R-CNN for detecting a single oil pad from an input image. The model is trained with input images of size 256x256 containing a single oil pad located at the center of the image. The model predicted an output image with a bounding box if an oil pad is present in the image. As initial work, this study showed that the Faster RCNN model is capable of detecting oil pads from aerial images. However, this study did not investigate the performance of the model while multiple oil pads with varying orientations and the presence of other visually similar features like rock, sand cleared areas. Further the model proposed by (Sunil et al., 2021) is limited to an input image size of 256x256 which cannot cover a large area with in a single image.

There can be several oil pads spread across a region, as oil extraction is important for Wyoming's economy. Hence it is important for the authorities to keep track of them. For real-world applications, it would be necessary for a model to detect multiple oil pads from aerial images irrespective of their location (oil pad located anywhere in the image). Figure 1 is an image of an oil pad taken from Google Earth. The objective of this study is to identify multiple oil pads irrespective of their orientation, presence of visually similar background features, and

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVI-M-2-2022-91-2022 | © Author(s) 2022. CC BY 4.0 License.

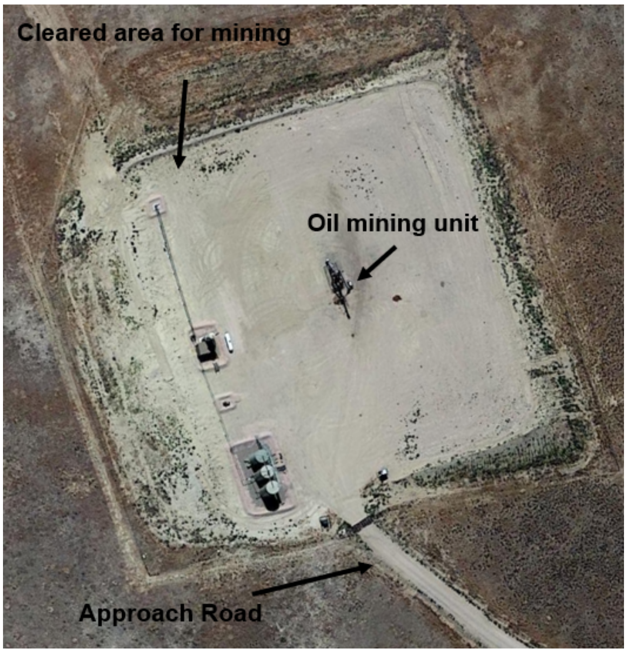

Figure 1. Example of oil pad from google earth.

varying position of the oil pad within the frame of the aerial images. In an aerial image the model must identify and return the locations of all oil pads present in the image marked by a bounding box. In this study, we trained a Faster R-CNN network with 1024x1024 input images which can detect multiple oil pads in any orientation from the provided input image.

## 2. MATERIALS AND METHODS

### 2.1 Multiple oil pad dataset

The images are downloaded from the Wyoming GIS Center, University of Wyoming, USA. The images in the dataset were captured by manned aircraft flown over Wyoming, USA. The captured images are of high spatial resolution (1m x 1m), containing 4 bands (Red, Green, Blue and infrared) and they have the longitude and latitude coordinates embedded in each pixel (Geo-tagged images). The images used in this study were acquired in 2017. These images contain multiple oil pads, man-made structures (storage tanks, approach roads and vehicles) and natural vegetation and soil cover.

### 2.2 Dataset preparation

The aerial images contain multiple oil pads, interconnecting roads and other structures such as abandoned oil pads, rock structures and shrubs. The required images were cropped from the large aerial images to images of size 1024x1024 with the help of GNU Image Manipulation Program GIMP software (The GIMP Development Team, 2019). Only the color bands (red, blue and green) is used in this study. The cropped images were split into the training set and the validation set. The training set contains 41 images containing 79 oil pads in total. The validation set contains 50 images with 47 oil pads in total.

### 2.3 Image Annotation

Assigning metadata to the image with the location of the target is called data annotation. Adding the metadata is important as it enables the model to learn about the desired features. The most commonly used annotations are bounding boxes, key points, lines and segmentation. Drawing bounding boxes on the aerial images is difficult as the targets is might be small and might not be clear or since the target might be located at the edge of the image. Structures will appear clearer in Google Earth than our dataset, as the pixel resolution of the images is better on Google Earth. The annotation is guided with help of Google Earth images, with the latitude and longitude from training images were used to verify the annotated region in training images. We looked up the coordinates on Google Earth, to confirm if the cleared area is an oil pad or not. The coordinates were extracted from the pixels of the image with ERDAS software (Saxena, 2015). On confirming the locations of the oil pad, we proceeded to annotate the oil pads in the input images. For this study, we used a rectangular bounding box, fitting the cleared area around the oil pad including the portion of the approach road. The annotation was done with the help of a tool called the VGG image annotator (VIA) (Dutta and Zisserman, 2019) tool. The annotation was completed as shown in Figure 2. The region inside the bounding box is considered as the region of interest (ROI). The training set was annotated and then passed to the model as input. The model learns from the ROI and can recognize the patterns when presented with an unlabelled test image. The annotation is also similarly done for the validation set to calculate the IoU score to evaluate the model performance.

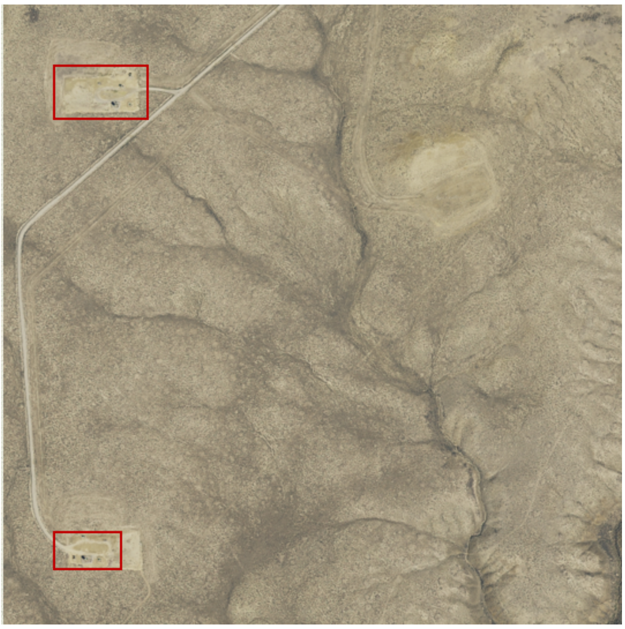

Figure 2. Input image with multiple oil pads used for training the model.

## 3. METHODOLOGY

### 3.1 Data Augmentation

There were 41 images containing 79 oil pads for training. In order to improve the performance of the model, data augmentation techniques were used. Random rotation augmentation, one of the geometric augmentation techniques, was used where the pixels of the input images were rotated to degrees: 90, 180 and 270. For each degree rotation, pixels in the images were rotated, hence the corresponding bounding boxes have to be adjusted.

By implementing the data augmentation techniques the number of images was increased to 164 and the number of oil pads in the training set was increased to 316.

### 3.2 Faster R-CNN for multiple oil pad detection

This network comes from the family of R-CNN-based (Girshick, 2015) (Girshick et al., 2013) object detection algorithms. Faster R-CNN is less time-consuming than prior versions, the Fast RCNN(Girshick, 2015). Faster R-CNN requires less time for processing data, and it is also more accurate than its previous versions. The presence of a region proposal network (RPN) in Faster R-CNN distinguishes it from earlier versions (Girshick, 2015). RPN is a proposal network used to predict where the target is present in the image. The convolutional neural network (CNN) receives the input image. The output is a feature map that corresponds to the input of the CNN. The feature map is passed on to the RPN layer which generates the appropriate anchors that helps the network to find where the oil pad is located in the image. The region with the oil pad is marked as foreground class and the remaining regions is considered as the background class by the network during training. The RPN layer is capable of finding the locations of the targets present anywhere in the image. Then using image classification and bounding box regression, the feature maps are categorized and the required features in the bounding boxes are learned by the model during training.

The Faster RCNN network in this study uses ResNet-50 (He et al., 2015) for the RPN layer. Anchor boxes of varying sizes were used as the size of the oil pad can vary. The training set to the model contains images with multiple oil pad and their corresponding annotations in a JSON file. Both the images and the JSON is given as input for training the model. The initial model was trained with the ResNet-50 pre-trained weights for 300 epochs with each epoch running for 1000 iterations. The best performing epoch was used as the initial weights for the transfer learning model, which was trained for 100 epochs with each epoch running for 1000 iterations. From the transfer learning model, the best preforming epoch was chosen based on the minimum loss averaged over randomized validation images.

### 3.3 Segregation of images into different categories

The validation set consists of 50 images which were split up into 3 categories based on the number of oil pads present in the images. The categories were: no oil pad, single oil pad and multiple oil pad. By segregating the validation set into different categories, analysis can be done on the performance of the model subjected to different conditions. The validation set was not seen by the model during training. For the no oil pad set, under perfect conditions, the trained model must not return any bounding boxes. For the single oil pad set, the oil pad can be located anywhere in the images. This is to verify if the trained model is capable of detecting the oil pad in any location. For the multiple oil pad set, there can be multiple oil pads in a single frame of the image with other structures. This is to see if the model is effectively able to detect multiple oil pads from other visually similar features.

After training the proposed model, the model performance was calculated using standard metrics. The standard metrics are precision (1), recall (2), F1 score (3) and average IoU (4). The metrics were calculated for each of the categories of the validation set. In the present work, there is only 1 target class: oil pad. The true positive (TP) indicates the number of oil pads that were correctly detected by the trained model. The false positive (FP) indicates the number of objects that were falsely identified by the trained model as an oil pad. The false-negative (FN) indicates the number of oil pads that were not detected by the network. The true negative (TN) is not being calculated as there is only 1 target.

$$Precision = \frac{TP}{(TP + FP)} \tag{1}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{2}$$

$$F1\,Score = \frac{(2 * Precision * Recall)}{(Precision + Recall)} \tag{3}$$

$$IoU = \frac{Area\,of\,Overlap}{Area\,of\,Union} \tag{4}$$

## 4. RESULTS AND DISCUSSION

### 4.1 Faster R-CNN Training

The initial model was trained for 300 epochs with pre-trained ResNet-50 weights that was downloaded before training. The 211th epoch had the least loss, was picked and performance was evaluated. The 211th epoch was used as initial weights for the transfer learning model and trained for 100 epochs with the remaining model parameters being kept the same. Epoch 1 of the transfer learning model returned with the least loss. The Epoch 1 model was evaluated on the validation images. For the validation the output of model with confidence level above 80% is taken

### 4.2 Validation Category

The categories contained 20, 20 and 10 numbers of images for no oil pad, single oil pad and multiple oil pad sets correspondingly. The model was evaluated for each category-wise and the result are tabulated.

Figure 3 and Figure 4 shows the model predictions for a few images of the validation set. The colored rectangular bounding boxes are the model's prediction of an oil pad. With the inclusion of data augmentation and transfer learning, the model results have been improved and the misclassifications have been reduced compared to the initial model without transfer learning. The initial model tends to misclassify the roads and other patches on the ground as oil pads. The present model tends to identify some of the cleared areas as oil pads (Figure 4). Some cleared area also tends to have the shape similar to that of the oil pad.

Figure 5 and Figure 6 shows the set of images from no oil pad set with only abandoned oil pads and the roads connecting them. On the other hand, the model was able to distinguish well between a working oil pad and an abandoned one. Model misclassified abandoned oil pads as active ones when green patches of vegetation in them (Figure 6). The results obtained were better in comparison to the initial model.

The model was able to detect 85% of the oil pad in single oil pad set and 76% of the oil pad from the multiple oil pad set.

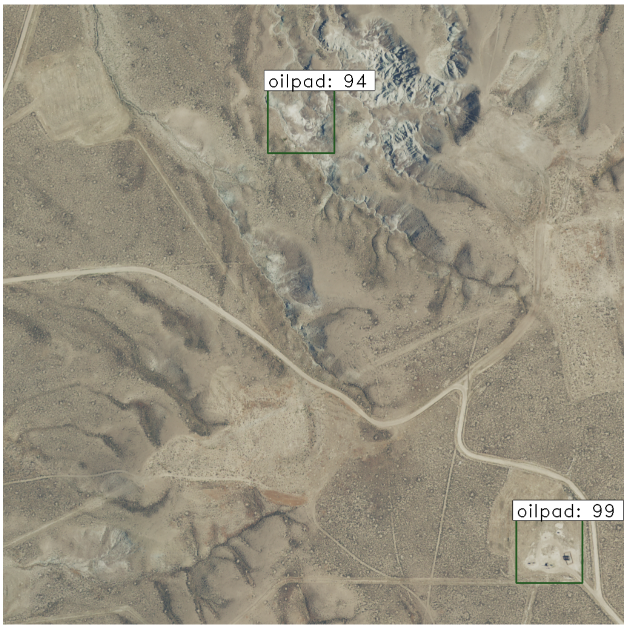

Figure 3. Model prediction on validation image from single oil pad set with 1 oil pad and rock structures.

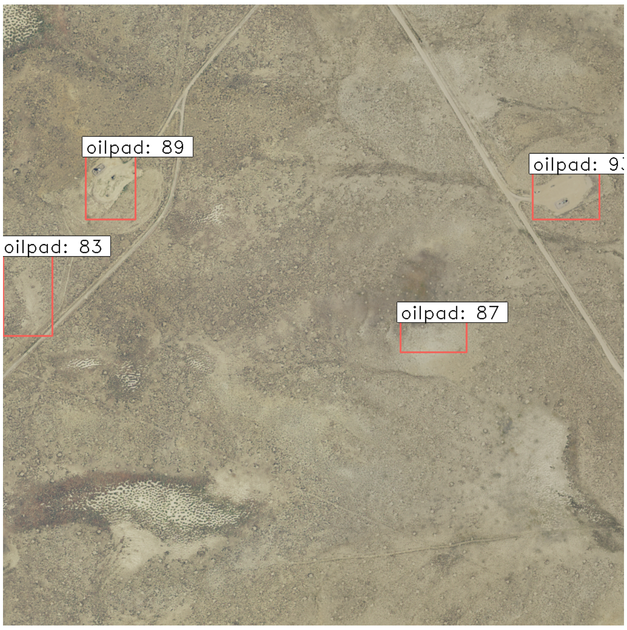

Figure 4. Model prediction on validation image from multiple oil pad set with 2 oil pads and naturally found cleared area.

| | Actual Class | |
|---|---|---|
| Predicted Class | Positive | Negative |
| Positive | 0 (TP) | 9 (FP) |
| Negative | 0 (FN) | nil |

Table 1. Confusion matrix for no oil pad set, to see if the model returns any false detections.

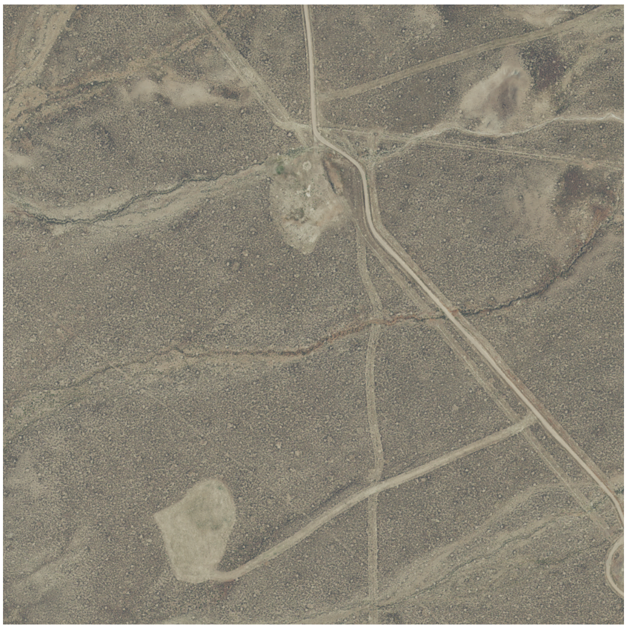

Figure 5. Model prediction on validation image from no oil pad set with abandoned oil pad and roads.

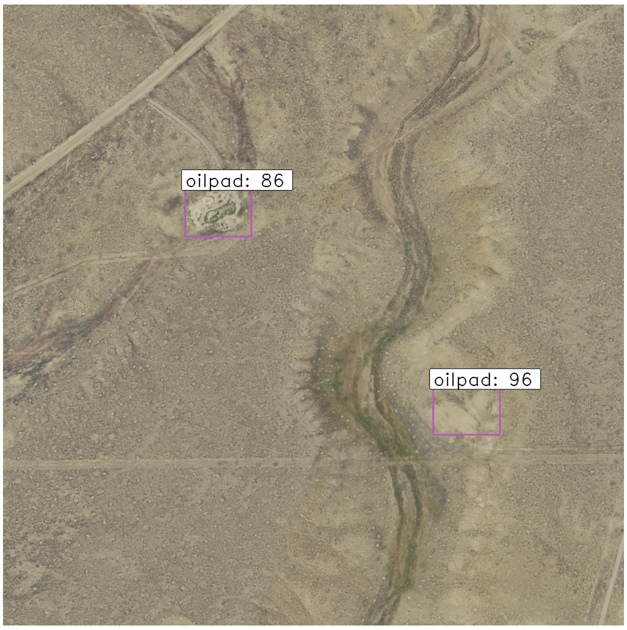

Figure 6. Model prediction on validation image from no oil pad set with abandoned oil pads with green patches and roads.

| | Actual Class | |
|---|---|---|
| Predicted Class | Positive | Negative |
| Positive | 17 (TP) | 9 (FP) |
| Negative | 3 (FN) | nil |

Table 2. Confusion matrix for single oil pad set, to see if the model is capable to detecting oil pad located anywhere in the image.

In Table 1, Table 2 and Table 3, the true negative (TN) class cannot be calculated as there is only oil pad as a target class in the ground truth. In Table 1, the model misclassified other structures as an oil pad. Even though there was no oil pad in those images, there were a few structures similar to the oil pad that mislead the model. The same is observed for the results in Table 2 and Table 3. The model tends to misclassify some

| | Actual Class | |
|---|---|---|
| Predicted Class | Positive | Negative |
| Positive | 20 (TP) | 9 (FP) |
| Negative | 6 (FN) | nil |

Table 3. Confusion matrix for multiple oil pad set,to see if the model is able to detect multiple oil pads effectively in a single frame.

| Test Category | Precision | Recall | F1 Score | Avg IoU |
|---|---|---|---|---|
| Single oil pad | 0.65 | 0.85 | 0.74 | 0.69 |
| Multiple oil pad | 0.68 | 0.76 | 0.72 | 0.65 |

Table 4. Precision, recall, F1 score and average IoU for single and multiple oil pad set.

| Categories | No of false detection |
|---|---|
| Cleared Area | 12 |
| Rock Structures | 5 |
| Sand Patterns and shrubs | 5 |
| Abandoned oil pad | 3 |
| Road Intersection | 2 |

Table 5. Precision, recall, F1 score and average IoU for single and multiple oil pad set.

structures that have visually similar features as an oil pad. The standard metrics for single and multiple oil pads obtained by the proposed method are tabulated in table 4. For both cases, the recall score is high, as very few of the oil pads were not identified by the model.

For a better understanding, all the false positives from the validation set are grouped in Table 5. Cleared area contributed to the most false positives, as the cleared area appears very similar to an oil pad. Some of these false positives contribute to the low precision scores for the single and multiple oil pad set.

## 5. CONCLUSION AND FUTURE WORK

Faster R-CNN model is capable of detecting multiple oil pads from aerial images with different shape, size and orientation. It is able to distinguish between a working oil pad and an abandoned one. The model also does not misclassify between road and oil pads by applying the transfer learning approach and data augmentation. With 65% and 68% precision scores, it indicates the potential of getting better results, upon reducing the number of false positives. With 85% and 76% recall scores, it indicates the model is able to identify oil pads from different scenarios. There were plenty of cleared area that look very similar to that of an oil pad which majorly mislead the model and contributed to the low precision scores.

The limitations mentioned above can be addressed in future work. The future work may include multi-class segmentation, moving away from the bounding box approach. Using the different categories that were identified by grouping the false positives together. By using these categories as target classes for training, we expect the model to learn the differences between each class during the training process. In turn reducing the number of false positives in the process.

## ACKNOWLEDGEMENTS

The authors thank UW Wyoming GIS center for providing the high spatial resolution true-color aerial images that were used in this study.

## REFERENCES

Ajay, A., Sowmya, V., Soman, K. P., 2017. Vehicle detection in aerial imagery using eigen features. *2017 International Conference on Communication and Signal Processing (ICCSP)*, 1620–1624.

Bah, M. D., Hafiane, A., Canals, R., 2018. Deep Learning with Unsupervised Data Labeling for Weed Detection in Line Crops in UAV Images. *Remote Sensing*, 10(11). https://www.mdpi.com/2072-4292/10/11/1690.

Dutta, A., Zisserman, A., 2019. The VGG Image Annotator (VIA). *CoRR*, abs/1904.10699. http://arxiv.org/abs/1904.10699.

Geetha, M., Manoj, M., Sarika, A. S., Mohan, M., Rao, S. N., 2017. Detection and estimation of the extent of flood from crowd sourced images. *2017 International Conference on Communication and Signal Processing (ICCSP)*, 0603–0608.

Girshick, R., 2015. Fast r-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2013. Rich feature hierarchies for accurate object detection and semantic segmentation.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2016. Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1), 142–158.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep residual learning for image recognition.

Ho, M.-J., Lin, Y.-C., Hsu, H.-C., Sun, T.-Y., 2019. An efficient recognition method for watermelon using faster r-cnn with post-processing. *2019 8th International Conference on Innovation, Communication and Engineering (ICICE)*, 86–89.

Jiao, L., Zhang, F., Liu, F., Yang, S., Li, L., Feng, Z., Qu, R., 2019. A Survey of Deep Learning-Based Object Detection. *IEEE Access*, 7, 128837–128868.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A. C., 2015. SSD: Single Shot MultiBox Detector.

Mo, N., Yan, L., 2020. Improved Faster RCNN Based on Feature Amplification and Oversampling Data Augmentation for Oriented Vehicle Detection in Aerial Images. *Remote Sensing*, 12(16), 2558.

Mohan, V. S., Sowmya, V., Soman, K. P., 2018. Deep neural networks as feature extractors for classification of vehicles in aerial imagery. *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, IEEE.

Osco, L. P., Junior, J. M., Ramos, A. P. M., de Castro Jorge, L. A., Fatholahi, S. N., de Andrade Silva, J., Matsubara, E. T., Pistori, H., Gonçalves, W. N., Li, J., 2021. A review on deep learning in UAV remote sensing. *International Journal of Applied Earth Observation and Geoinformation*, 102, 102456.

Redmon, J., Farhadi, A., 2017. Yolo9000: Better, faster, stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6517–6525.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks.

Saxena, P., 2015. GEO-LOCATION BASED AUGMENTED REALITY APPLICATION. *International Journal of Research in Engineering and Technology*, 04, 495-498.

Sunil, A., Sajithvariyar, V., Vishvanathan, S., Sivanpillai, R., Kp, S., 2021. Identifying oil pads in high spatial resolution aerial images using faster r-cnn. XLIV-M-3-2021, 155–161.

The GIMP Development Team, 2019. https://www.gimp.org.